In [2]:
```
from textblob import TextBlob
from wordcloud import WordCloud
import pandas as pd
import numpy as np
import re
import matplotlib.pyplot as plt
%matplotlib inline
```

In [3]:
```
df = pd.read_csv('Reviews.csv')
```

In [15]:
```
from wordcloud import WordCloud
from wordcloud import STOPWORDS
```

In [4]:
```
df.head()
```

Out[4]:

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominat
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	

In [5]:
```
df.shape
```

Out[5]:
```
(568454, 10)
```

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 568454 entries, 0 to 568453
Data columns (total 10 columns):
 #   Column                  Non-Null Count   Dtype 
---  ------                  --------------   ----- 
 0   Id                      568454 non-null  int64 
 1   ProductId               568454 non-null  object
 2   UserId                  568454 non-null  object
 3   ProfileName             568438 non-null  object
 4   HelpfulnessNumerator    568454 non-null  int64 
 5   HelpfulnessDenominator  568454 non-null  int64 
 6   Score                   568454 non-null  int64 
 7   Time                    568454 non-null  int64 
 8   Summary                 568427 non-null  object
 9   Text                    568454 non-null  object
dtypes: int64(5), object(5)
memory usage: 43.4+ MB
```

In [7]:
```
df.Summary.head()
```

Out[7]:
```
0    Good Quality Dog Food
1        Not as Advertised
2    "Delight" says it all
3           Cough Medicine
4              Great taffy
Name: Summary, dtype: object
```

In [8]:
```
df.Text.head()
```

Out[8]:
```
0    I have bought several of the Vitality canned d...
1    Product arrived labeled as Jumbo Salted Peanut...
2    This is a confection that has been around a fe...
3    If you are looking for the secret ingredient i...
4    Great taffy at a great price.  There was a wid...
Name: Text, dtype: object
```

In [9]:
```
from nltk.corpus import stopwords
from textblob import TextBlob
from textblob import Word
```

In [10]:
```
df['Text'] = df['Text'].apply(lambda x: " ".join(x.lower() for x in x.split ()))
```

In [11]:
```
df['Text'] = df['Text'].str.replace('[^\w\s]'," ")
df.Text.head(5) #removing punctuations
```

```
C:\Users\ashaikh\AppData\Local\Temp\ipykernel_15616\3336156060.py:1: FutureWarn
ing: The default value of regex will change from True to False in a future vers
ion.
  df['Text'] = df['Text'].str.replace('[^\w\s]'," ")
```

Out[11]:
```
0    i have bought several of the vitality canned d...
1    product arrived labeled as jumbo salted peanut...
2    this is a confection that has been around a fe...
3    if you are looking for the secret ingredient i...
4    great taffy at a great price  there was a wide...
Name: Text, dtype: object
```

In [12]:
```
from nltk.corpus import stopwords
```

In [13]:
```
stop = stopwords.words('english')
df['Text'] = df['Text'].apply(lambda x: " ".join(x for x in x.split() if x not in
df.Text.head() #words like i, the, am... have been removed
```

Out[13]:
```
0    bought several vitality canned dog food produc...
1    product arrived labeled jumbo salted peanuts p...
2    confection around centuries light pillowy citr...
3    looking secret ingredient robitussin believe f...
4    great taffy great price wide assortment yummy ...
Name: Text, dtype: object
```

In [14]:
```
df['Text'] = df['Text'].apply(lambda x: " ".join([Word(word).lemmatize() for word
df.Text.head()
```

Out[14]:
```
0    bought several vitality canned dog food produc...
1    product arrived labeled jumbo salted peanut pe...
2    confection around century light pillowy citrus...
3    looking secret ingredient robitussin believe f...
4    great taffy great price wide assortment yummy ...
Name: Text, dtype: object
```

In [16]:
```
df.columns
```

Out[16]:
```
Index(['Id', 'ProductId', 'UserId', 'ProfileName', 'HelpfulnessNumerator',
       'HelpfulnessDenominator', 'Score', 'Time', 'Summary', 'Text'],
      dtype='object')
```

In [17]:
```
df.Score.value_counts()
```

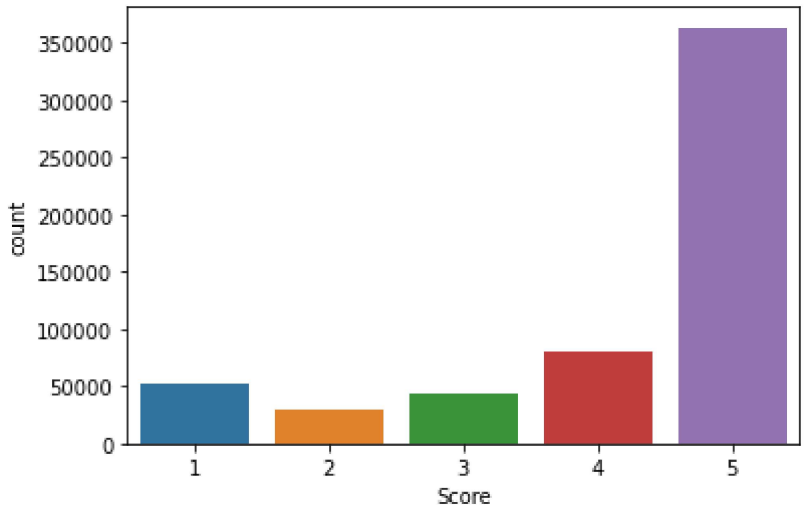
Out[17]:
```
5    363122
4     80655
1     52268
3     42640
2     29769
Name: Score, dtype: int64
```

In [18]:
```
#4-5 shows the positive reviews
#1-2 shows the negative reviews
#3 neutral reviews
```

In [19]:
```
import seaborn as sns
```

In [20]:
```
sns.countplot(data = df, x= 'Score')
```

Out[20]:
```
<AxesSubplot:xlabel='Score', ylabel='count'>
```



In [21]:
```
reviews = df #dataframe eda
reviews.dropna(inplace=True)
```

In [22]:
```
score_1 = reviews[reviews['Score'] == 1]
score_2 = reviews[reviews['Score'] == 2]
score_3 = reviews[reviews['Score'] == 3]
score_4 = reviews[reviews['Score'] == 4]
score_5 = reviews[reviews['Score'] == 5] #storing values, multiple strings
```

In [23]:
```
reviews_sample = pd.concat([score_1,score_2,score_3,score_4,score_5],axis=0)
reviews_sample.reset_index(drop=True,inplace=True) #storing values, single string
```

In [24]:
```
negative_reviews = reviews_sample[reviews_sample['Score'].isin([1,2])]
positive_reviews = reviews_sample[reviews_sample['Score'].isin([4,5])]
negative_reviews_str = negative_reviews.Summary.str.cat()
positive_reviews_str = positive_reviews.Summary.str.cat()
```

In [54]:
```
wordcloud_negative = WordCloud(background_color='white').generate(negative_review
wordcloud_positive = WordCloud(background_color='white').generate(negative_review
fig = plt.figure(figsize=(10,10))
ax1 = fig.add_subplot(211)
ax1.imshow(wordcloud_negative,interpolation='bilinear')
ax1.axis("off")
ax1.set_title("Negative Reviews", fontsize=20)
```

Out[54]: Text(0.5, 1.0, 'Negative Reviews')

In [56]:
```
fig = plt.figure(figsize=(10,10))
ax1 = fig.add_subplot(211)
ax1.imshow(wordcloud_positive,interpolation='bilinear')
ax1.axis("off")
ax1.set_title("Positive Reviews", fontsize=20)
```

Out[56]: Text(0.5, 1.0, 'Positive Reviews')

In [31]:
```
!pip install vaderSentiment
```

```
Requirement already satisfied: vaderSentiment in c:\users\ashaikh\anaconda3\lib
\site-packages (3.3.2)
Requirement already satisfied: requests in c:\users\ashaikh\anaconda3\lib\site-
packages (from vaderSentiment) (2.27.1)
Requirement already satisfied: idna<4,>=2.5 in c:\users\ashaikh\anaconda3\lib\s
ite-packages (from requests->vaderSentiment) (3.3)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\ashaikh\anaconda3
\lib\site-packages (from requests->vaderSentiment) (2021.10.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\ashaikh\anacon
da3\lib\site-packages (from requests->vaderSentiment) (1.26.9)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\ashaikh\an
aconda3\lib\site-packages (from requests->vaderSentiment) (2.0.4)
```

In [32]:
```
import seaborn as sns
import re
import os
import sys
import ast
plt.style.use('fivethirtyeight')
cp = sns.color_palette()
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

In [34]:
```
emptyline =[]
for row in df['Text']:
    vs=analyzer.polarity_scores(row)
    emptyline.append(vs)
df.sentiments=pd.DataFrame(emptyline)
df.sentiments.head()
```

```
C:\Users\ashaikh\AppData\Local\Temp\ipykernel_15616\3505471067.py:5: UserWarnin
g: Pandas doesn't allow columns to be created via a new attribute name - see ht
tps://pandas.pydata.org/pandas-docs/stable/indexing.html#attribute-access (http
s://pandas.pydata.org/pandas-docs/stable/indexing.html#attribute-access)
  df.sentiments=pd.DataFrame(emptyline)
```

Out[34]:

	neg	neu	pos	compound
0	0.000	0.503	0.497	0.9413
1	0.129	0.762	0.110	-0.1027
2	0.130	0.587	0.283	0.8532
3	0.000	0.854	0.146	0.4404
4	0.000	0.369	0.631	0.9468

In [36]:
```
df_c = pd.concat([df.reset_index(drop=True), df.sentiments], axis=1)
df_c.head() #merging sentiments back to review data
```

Out[36]:

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominat
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	

In [37]:
```
df_c['Sentiments'] = np.where(df_c['compound'] >= 0 , 'Positive','Negative')
df_c.head()
```

Out[37]:

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominat
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	

In [40]:
```
result=df_c['Sentiments'].value_counts()
result.plot(kind="bar", rot=0, color=['black','red'])
```

Out[40]: <AxesSubplot:>